REMARKS

Applicant respectfully requests reconsideration and allowance of the subject application.

Claims 1-77 were originally submitted.

Claims 58-66 and 76-77 have been cancelled without prejudice.

Claims 1-57 and 67-75 have been previously amended.

Claims 1, 20, and 39 are currently amended.

Claims 1-57 and 67-75 are pending.

35 U.S.C. §103

Claims 1-12, 14-31, 32-50, 52-57 and 67-75 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,430,707 to Matthews et al. (Matthews) in view of U.S. Patent No. 6,633,876 B1 to Heatlie (Hcatlie). Applicant respectfully traverses the rejection.

Matthews teaches a client station on a computer network that uses an operating system, such as JavaOS. The operating system is permanently stored at a server computer on the computer network, rather than on storage media at the client location. The operating system is loaded and installed at the client computer upon bootup of the client computer. If a malfunction causes the client computer to go off-line (i.e., disappear from the computer network), a dump image is sent to the server computer before the client reboots, so that the cause of failure can be determined. The dump image includes the contents of memory, including the operating system, and the code, stack, threads, registers, and local/global variables of the client computer. The dump image is sent to the server computer in formatted packets. At the server computer, the formatted packets making up the

dump image are stored on storage media, and the dump image is reformatted for use by a standard debugging tool. (Abstract of Matthews, and Matthews, col. 2 lines 21-40).

To ensure a remote dump capability under all conditions a JavaOS client application in the client computer includes a secondary means of communication with the server computer, that exists independent of the normal operating mechanism of the client computer (i.e., independent of the TCP/IP communication stack). The system taught in Matthews, relies on a JavaOS scheme that provides for a preboot execution (PXE) boot process to complete, and have the system return memory occupied by a user datagram protocol (UDP) stack to a system heap. Typically, a PXE boot process discards the UDP stack once the boot process is complete. In order to utilize this UDP stack for later system dumps to the server, the JavaOS (i.e., operating system) of the client computer, reserves the UDP stack subsequent to completion of the boot process instead of reallocating it to the general system heap. (Matthews, col. 4 lines 15-37).

Heatlie in its background section, teaches related art that suggests when a computer system crashes an exception handling routine typically saves post-mortem information specifying the state of the computer system after the failure to a crash dump file. This crash dump file typically contains much of the contents of the memory of the computer system immediately after the failure, including the state of various threads and the contents of various buffers. By viewing this crash dump file, an engineer is often able to diagnose the cause of the computer system failure. (Heatlie, col. 1 lines 16-24).

16   MS#147674.01.filename MSI-4081/S.htm

Independent claim 1 recites "[a] method for generating a dump file, the

method comprising:

a. generating a minidump file that does not include all volatile
system memory containing at least:

i. thread information for at least one running thread,
ii. context information for the thread,
iii. callstack information for the thread,
iv. process information for a process in which the thread is
running, and
v. information identifying a reason comprising one of the
following reasons: callstack fault, processor fault, and application
program fault, for generating the minidump file; and

b. storing the minidump file to a storage medium.

Claim 1 recites "generating a minidump file that does not include all

volatile system memory containing at least ...". Matthews does not teach or

suggest this element. In particular, the recited element is contrary to Matthews,

which teaches specifically that the dump image includes "*the* contents" of

memory. There is no indication that only part of the memory might be dumped.

Rather, the language "the contents" indicates that all of the memory is included in

the dump image. This is particularly provided in order for the *server* to determine

the cause of the failure. Therefore, Matthews fails to teach or suggest "minidump

file that does not include all of all volatile system memory". The rejection of

claim 1 is unjustified for this reason.

Claim 1 also recites that the minidump file contains "information

identifying a reason ... for generating the minidump file." The Action admits that

Matthews fails to teach this element, and instead looks to the teaching of Heatlie.

In particular, the Action argues that "Heatlie teaches information identifying a

reason comprising one of the following reasons: callstack fault, processor fault,

and application fault, for generating the minidump file." In Heatlie, however, actual identification of the reason is performed by a server or some other computer, *as a result of examining the dump file*. The reason is not contained in the dump file itself. Thus, Heatlie does not teach a dump file "containing" the reason. The rejection of claim 1 is unjustified for this further reason.

Heatlie teaches that when a computer (computer system) crashes or fails, post-mortem information is saved that specifies the state of the computer system after the failure. There is no teaching or suggestion that the post-mortem information includes a *reason* for generating post-mortem information. Although the state of the computer system is provided, additional analysis is expected to be made, "post-mortem," as to the reason for generating the state information. Claim 1, however, recites that the mini-dump file itself "contains" the reason. Heatlie does not teach this, and the rejection of claim 1 is unfounded for this additional reason.

Therefore, it would not have been obvious to combine the teaching of Matthews with Heatlie. Accordingly, a combination of Matthews and Heatlie is improper. Applicant respectfully requests that the §103 rejection of claim 1 be withdrawn.

Dependent claims 1-12, 14-19, and 67-69 depend on claim 1, and are allowable at the least by virtue of their dependency on base claim 1. Applicant respectfully requests that the §103 rejection of claims 1-19 and 67-69 be withdrawn.

Independent claim 20 recites "[a] computer-readable medium having computer-executable instructions for causing at least one processor to perform acts comprising:

gathering minidump file information that does not include all volatile system memory but does include at least thread information for at least one running thread, context information for the thread, callstack information for the thread, process information for the process in which the thread is running, and information identifying a reason comprising one of the following reasons: callstack fault, processor fault, and application program fault, for generating the minidump file.

The Action rejects claim 20 based on similar reasons as claim 1. Applicant presents the arguments in support of claim 1, in support of claim 20. Accordingly, Applicant respectfully request that the §103 rejection of claim 22 be withdrawn.

Dependent claims 21-31, 33-38, 70, and 71 depend on claim 20, and are allowable at the least by virtue of their dependency on base claim 20. Applicant respectfully requests that the §103 rejection of claims 21-31, 33-38, 70, and 71 be withdrawn.

Independent claim 39 recites "[a]n apparatus comprising:

memory;

a data storage drive configured to write data files to at least one data storage medium; and

at least one processor operatively coupled to the memory and the data storage drive and configured to:

a. generate a minidump file that does not include all volatile system memory containing at least:
i. thread information for at least one running thread,
ii. context information for the thread,
iii. callstack information for the thread,
iv. process information for the process in which the thread is running, and
v. information identifying a reason comprising one of the following reasons: callstack fault, processor fault, and application program fault, for generating the minidump file, and

19

b. store the minidump file to the storage medium.

The Action rejects claim 39 based on similar reasons as claim 1. Applicant presents the arguments in support of claim 1, in support of claim 39. Accordingly, Applicant respectfully request that the §103 rejection of claim 39 be withdrawn.

Dependent claims 40-50, 52-57, and 73-75 depend on claim 39, and are allowable at the least by virtue of their dependency on base claim 39. Applicant respectfully requests that the §103 rejection of claims 40-50, 52-57, and 73-75 be withdrawn.

Claims 13, 32 and 51 are rejected under 35 U.S.C. 103(a) as being unpatentable over Matthews in view of Heatlie, and in further view of U.S. Patent No. 5,603,033 to Joannin (Joannin). Applicant respectfully traverses the rejection.

Claim 13 depends on claim 1, and as such includes the elements:

a. generating a minidump file that does not include all volatile system memory containing at least:

    i. thread information for at least one running thread,
    ii. context information for the thread,
    iii. callstack information for the thread,
    iv. process information for a process in which the thread is running, and
    v. information identifying a reason comprising one of the following reasons: callstack fault, processor fault, and application program fault, for generating the minidump file; and

b. storing the minidump file to a storage medium.

As discussed above, Matthews in view of Heatlie fails to teach or suggest these elements. The Action admits that Matthews and Heatlie do not teach "upon re-initialization, after having stored the minidump file to the storage medium, accessing the minidump file on the storage medium and using at least a portion of

the minidump to further understand an exception that was at least one reason for generating the minidump file" and looks to Joannin as teaching this element.

Joannin provides no assistance in light of Matthews and Heatlie as to the recited method of claim 13. Since Matthews and Heatlie do not teach the elements discussed above, the teaching of Ullman does not help.

Accordingly, a combination of Matthews, Heatlie, and Joannin is improper. Applicant respectfully requests that the §103 rejection of claim 13 be withdrawn.

Claims 32 and 51 are rejected based on similar reasons as claim 13. Applicant presents the arguments in support of claim 13, in support of claims 32 and 51. Accordingly, Applicant respectfully request that the §103 rejection of claims 32 and 51 be withdrawn.

CONCLUSION

All pending 1-57 and 67-75 are in condition for allowance. Applicant respectfully requests reconsideration and prompt issuance of the subject application. If any issues remain that prevent issuance of this application, the Examiner is urged to contact the undersigned attorney before issuing a subsequent Action.

Respectfully Submitted,

Dated: _9/30/05_   By: _____

Emmanuel A. Rivera
Reg. No. 45,760
(509) 324-9256 ext. 245